https://doi.org/10.2352/ISSN.2470-1173.2018.13.IPAS-193

Deep p-Fibonacci scattering networks

F. Battisti [a], *M. Carli* [a], *E. De Paola* [a], *and K. Egiazarian* [b];
[a] *Roma Tre University, Rome, Italy;*
[b] *Tampere University of Technology, Tampere, Finland*

Abstract

Recently, the use of neural networks for image classification has become widely spread. Thanks to the availability of increased computational power, better performing architectures have been designed, such as the Deep Neural networks. In this work, we propose a novel image representation framework exploiting the Deep p-Fibonacci scattering network. The architecture is based on the structured p-Fibonacci scattering over graph data. This approach allows to provide good accuracy in classification while reducing the computational complexity. Experimental results demonstrate that the performance of the proposed method is comparable to state-of-the-art unsupervised methods while being computationally more efficient.

Introduction

The availability of low cost digital imaging devices allows end users to easily act as content creators at almost no cost. The acquired images are often shared among friends through some communication network. Recent studies estimate that more than 250 billion photos have been uploaded to Facebook thus resulting in 350 million photos per day [1]. To manage this huge amount of data, new tools and dedicated representation frameworks are needed.

One of the basic tool is the automatic image classification that is fundamental in many fields, such as handwriting recognition, object identification, biometrics, etc. This task is usually performed by extracting features from the images and creating clusters based on some similarity criteria. The features extracted from the image to be classified are compared with the relevant features of each cluster: the image will belong to the cluster showing the lower difference with respect to the extracted features. Depending on the interaction between the user and the system during the classification, two types of classification can be identified: supervised and unsupervised.

One of the problem affecting the performances of image classification is the possible rigid rotation and/or translation of the objects in the scene. The content variability does not affect the real information of the scene, however, it heavily affects the similarity computation between features extracted from the image under test and reference features.

In this work the problem of image classification by means of Deep Neural networks is addressed. In more details, a theoretical analysis of p-Fibonacci scattering network is performed and is used as basis for representing the images under test. The main goal is to extend the method proposed in [2] by exploiting the Fibonacci p-sequences in the scattering procedure thus improving the classification performances. The contributions of this work may be summarized as follows:

- a novel classification algorithm is proposed based on an unsupervised learning approach;
- the scattering procedure is obtained by exploiting the p-Fibonacci scattering that, based on the selected p-sequence, allows more flexible selection of the input coefficients;
- the p-Fibonacci based decomposition results in a sparse representation of data that allows computational complexity and storage savings. Furthermore, as generally observed in sparse representations, there is a lower probability of having dependencies due to noise in the data representation.

p-Fibonacci sequences

The p-Fibonacci sequence, the generalization of famous Fibonacci sequence proposed by Leonardo da Pisa in [3], can be defined as follows. Given $(p,n) \in \mathbb{N}$, the sequence of p-Fibonacci numbers $F_{(p)}(n)$ is:

$$F_p(n) = \begin{cases} 0 & \text{if } n < 0 \\ 1 & \text{if } n = 0 \\ F_p(n-1) + F_p(n-p-1) & \text{if } n > 0 \end{cases}$$

The Zeckendorf theorem [4] states that every positive integer m can be represented uniquely as the sum of one or more distinct p-Fibonacci numbers in the sequence $\{a_n\}$, $a_n = F_p(n)$, for $n \geq 1$, in such a way that the sum does not include any two consecutive p-Fibonacci numbers.

A useful representation of the p-Fibonacci numbers is obtained by exploiting the binary p-Fibonacci tree that is a non-oriented graph in which each node has outdegree of at most two. In general, if there is a path between two nodes a and b, a is called *predecessor* of b and b *successor* of a. The *depth* of a node is the length of the path from the root to that node. If $depth(b) = depth(a) + 1$ it is said that b is a *child* of a and a is a *parent* of b.
In the Fibonacci case, the root is indexed by $F_p(n) = N$, its sons are $F_p(n-1)$ and $F_p(n-1-p)$. This recursion is repeated for each node until the last level which is composed by N nodes. Let $N = F_p(n)$ be the leaves of the binary tree T; at the bottom level, the leaves are clustered in three sets: w_0, w_1, and w_2, where

- $w_0 = \{\, \alpha_0, \cdots, \alpha_n \mid \alpha_i \text{is a left outedge} \,\}$
- $w_1 = \{\, \beta_0, \cdots, \beta_n \mid \beta_i \text{ is a right outedge} \,\}$
- $w_2 = \{\, \gamma_0, \cdots, \gamma_k \mid \gamma_j \text{ is a single child} \,\}$

The cardinality of w_2 is $|w_2| = F_p(n-1) - F_p(n-1-p)$. It can be easily demonstrated that the cardinality of w_0 and w_1 is $F_p(n-1-p)$.

The core element of the scattering transform is the p-Fibonacci matrix that can be obtained through the p-Fibonacci binary tree. Starting from the root of the p-Fibonacci tree, let us label each edge of the tree as follows:

1. If the node has two children, the left outedge will be denoted with 0 and the right one with 1.
2. If the node has only one child, the outedge will have label 0.

Each node of the binary tree may be indexed by a vector $(\alpha_1(a), ..., \alpha_k(a))$ $(\alpha_j \in \{0,1\},\ j = 1, ..., k)$, where k is the node depth and α_j are labels of the edges to that node starting from the root of the tree. Each possible path $F_p(n)$ represents the row of a matrix of size $F_p(n) \times F_p(n-1)$ of elements 0 or 1. The p-Fibonacci matrix can be obtained through recursion:

$$H_{F_p(n)} = \begin{pmatrix} a & H_{F_p(n-1)} \\ b & H_{F_p(n-1-p)} \end{pmatrix}$$

where a is a column of $F_p(n-1)$ zeros and b is a column of $F_p(n-1-p)$ ones, $H_{F_p(n-1)}$ and $H_{F_p(n-1-p)}$ are the matrices corresponding to $F_p(n-1)$ and $F_p(n-1-p)$ and finally b is a $F_p(n-1-p) * p$ matrix of zeros.

Proposed method

The proposed technique allows to represent an image by exploiting a scattering transform based on the p-Fibonacci decomposition. This transform results in a translation and rotation invariant representation of the data. Next, a supervised classification using a Gaussian kernel Support Vector Machine is applied for pooling the extracted features.

p-Fibonacci scattering transform

In [2], an algorithm for signal representation based on a connectivity graph is introduced. The input layer of the graph is a positive d-dimensional signal $x \in (\mathbb{R}^+)^d$. Let $x^{(j)}$ be the generic network layer at depth j. A deep neutral networks computes $x^{(j+1)}$ by applying a linear operator H_j to $x^{(j)}$. Due to the linearity property, the distance is preserved, up to a constant normalization factor λ so

$$\left|\left|H_j y - H_j y^{'}\right|\right| = \left|\left|H_j (y - y^{'})\right|\right| = \lambda \left|\left|y - y^{'}\right|\right|.$$

Applying H_j to $x^{(j)}$ the next network layer will be:

$$x^{(j+1)} = \left|H_j x^{(j)}\right|.$$

This operation is iterated up to a maximum level J to finally compute the last level $x^{(J)}$. This transformation H_j, specified by the multi-layer pairings π_j, is shown in Figure 1.

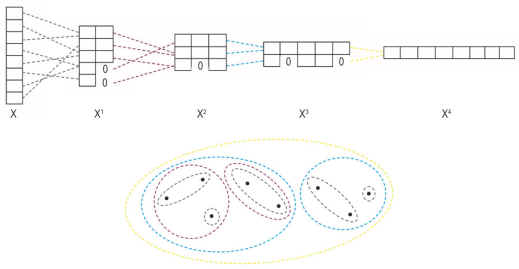

Figure 1. *Hierarchical partition of a graph.*

The maximum level J must be

$$J \leq n - p$$

where $d = F_p(n)$. The operator H_j applies the two bijective transformations:

$$(\alpha, \beta) \mapsto (\alpha + \beta, |\alpha - \beta|) = (u, v). \tag{1}$$

where $\alpha \in w_0$ and $\beta \in w_1$ and

$$\gamma \mapsto \gamma \tag{2}$$

where $\gamma \in w_2$. Since this is an invariant permutation, α, β, and γ may be recovered as follows:

$$\left(\frac{u+v}{2}, \frac{u-v}{2}\right) = (\max(\alpha,\beta), \min(\alpha,\beta)) \quad (3)$$

Furthermore, it is a reversible algorithm thus allowing to identify the initial array x starting from the last level by applying the inverse formulas (Eq. 3).

In this contribution we extend the approach proposed in [2] by exploiting the p-Fibonacci sequences for pairing the elements. In more details, the number of rows of x is a p-Fibonacci number $N = F_p(n)$. The procedure is based on i) the identification of the set w_0, w_1, and w_2, followed by ii) the application of the bijective transformation defined in Eq. (1) to $x(w_0)$ and $x(w_1)$, and the one defined in Eq. (2) to the set $x(w_2)$. The application of $|H_j|$ to $x^{(j)}$ allows to compute the next layer $x^{(j+1)} = \left|H_j x^{(j)}\right|$, obtained by re-grouping of the coefficients of $x^{(j)} \in \mathbb{R}^d$ into $F_p(n-1-p)$ pairs and without modifying the other part. This pairing is an invariant permutation $\pi_j = \{\pi_j(w_0), \pi_j(w_1)\}$:

$$x^{(j+1)}(w_0) = x^{(j)}(\pi_j(w_0)) + x^{(j)}(\pi_j(w_1)) \quad (4)$$

$$x^{(j+1)}(w_1) = \left|x^{(j)}(\pi_j(w_0)) - x^{(j)}(\pi_j(w_1))\right| \quad (5)$$

$$x^{(j+1)}(w_2) = x^{(j)}(w_2) \quad (6)$$

The pairing π_j specifies which index $\pi_j(w_0)$ is paired with $\pi_j(w_1)$. However, the ordering index w is not relevant, as w specifies the storing position in $x^{(j+1)}$ of the transformed values. The output $x^{(J)}$ is computed with $F_p(n-1-p) \times J$ additions and subtractions. It is worth noticing that $p = 0$ is the classical Haar representation $d = 2^n$, where $J \geq 0$ represents the maximum depth and $2^J \leq d$, and $|w_0| = |w_1| = \frac{d}{2}$. The classical Haar functions are defined by the dyadic splitting of time interval, resulting in an empty w_2 set.

If $p \geq 1$, $d = F_p(n)$, $J \geq 0$ is the maximum level and it must be $J \leq n - p$. As stated before, this is an orthogonal p-Fibonacci scattering $x^{(J)}$ because it applies, at each step, an orthogonal matrix to x, which depends on x and J. Indeed, for any $J \geq 0$ and any $(x,y) \in \mathbb{R}^{2d}$

$$\left|\left|x^{(J)} - y^{(J)}\right|\right| \leq ||x-y||. \quad (7)$$

Moreover, $x^{(J)} = DM_{x,J}x$, where $M_{x,J}$ is an orthogonal matrix which depends on x and J, D is a $F_p(n) \times F_p(n)$ diagonal matrix where the diagonal vector assumes $\frac{1}{\sqrt{2}}$ in the rows indexed by w_0 and w_1, and 1 in the rest. The result of the product is an orthogonal matrix, and

$$\left|\left|x^{(J)}\right|\right| \leq ||x||.$$

From Eq. (3) it is possible to recover α and β. It can be proved that the signal x can be reconstructed from 2^J distinct orthogonal p-Fibonacci scattering transforms, computed with different pairings π_j at each layer. That is, there exist 2^J different transforms such that all $x \in \mathbb{R}^d$ can be reconstructed from the coefficients of these 2^J transforms.

Let us denote by V the set of $d = F_p(n)$ vertices of a graph. The input layer is

$$x^{(0)}(w,0) = x(w).$$

The resulting $x^{(1)}(w,q)$ is an array of size $F_p(n-1) \times 2$, where in the first column there are the even Fibonacci numbers (those having 0 at the least significant bit in representation) and in the second the odd Fibonacci numbers (1 in the least significant bit). The even Fibonacci numbers derive from sums and the single child, while the odd Fibonacci numbers derive from the differences. The index $w \in \{1,2,...,F_p(n-1)\}$ specifies each pair of nodes paired by π_0 and $k \in \{0,1\}$ indicates if the coefficient is computed by a subtraction or a sum. If the coefficients derive from a subtraction, then k is set to 1, otherwise $k = 0$.

The next π_1 is obtained by pairing $F_p(n-1-p)$ pairs, that is

$$\pi_1 = (\pi_1(w_0), \pi_1(w_1))_{0 \leq w \leq F_p(n-1)}.$$

The coefficients $\left\{x^{(1)}(w,k)\right\}_w$ are thus paired for $k = 0$ or $k = 1$.

It can be shown that $x^{(j)}(w,k)$ is an array of size $F_p(n-j) \times F_p(p+j)$. For each $j \geq 0$, $w \in \{1,2,...,F_p(n-j)\}$ is a *spatial* index of a set of $V_{j,w}$ of $F_p(n)$ graph vertices, while $k \in \{0,...,F_p(p+j)-1\}$ indicates different p-Fibonacci scattering coefficients computed from the values of x in $V_{j,w}$. Each $x^{(j+1)}$ is computed by calculating a pairing π_j of the $F_p(n-j)$ rows of $x^{(j)}$. The row pairing

$$\pi_j = \left\{\left(\pi_j(w_0), \pi_j(w_1)\right)\right\}_{0 \leq w \leq F_p(n-j)},$$

pairs each $(\pi_j(w_0),k)$ with $(\pi_j(w_1),k)$ for $0 \leq k \leq F_p(p+j)$. Let v_0, v_1 and v_2 be the three sets of $F_p(p+j)$, which are used to index the column of $x^{(j+1)}$.

If $p = 0$, then $v_2 = \emptyset$ and by applying

$$(\alpha,\beta) \mapsto (\alpha+\beta, |\alpha-\beta|)$$

to each element gives

$$x^{(j+1)}(w,v_0) = x^{(j)}(\pi_j(w_0),k) + x^{(j)}(\pi_j(w_1),k) \quad (8)$$

and

$$x^{(j+1)}(w,v_1) = \left|x^{(j)}(\pi_j(w_0),k) - x^{(j)}(\pi_j(w_1),k)\right| \quad (9)$$

where $k \in \{0,1,\cdots,F_p(p+j)-1\}$.

For $p \geq 1$, $v_2 \neq \emptyset$: therefore to compute $x^{(j+1)}$ a different procedure holds. In more details: at level zero, $k=0$, since $x = x^{(0)}$ there are not subtractions. When $j=1$ $k \in \{0,1\}$, in the first column there are the sums and single child and in the second the differences are computed. It follows that

$$x^{(1)}(w,0) = x^{(0)}(\pi_1(w_0),0) + x^{(0)}(\pi_1(w_1),0)$$

or

$$x^{(1)}(w,0) = x^{(0)}(w_2,0)$$

and

$$x^{(1)}(w,1) = \left| x^{(0)}(\pi_1(w_0),0) - x^{(0)}(\pi_1(w_1),0) \right|.$$

For $j \geq 2$ the matrix $x^{(j+1)}$ is obtained by inserting in the column indexed by the elements of v_0 the sums and the only child of the rows of $x^{(j)}$, in the column indexed by the elements of v_1 the differences. The remaining columns of $x^{(j+1)}$ correspond to the columns of $x^{(j)}$ without any transformations. That columns of $x^{(j)}$ are kept equal for p levels and at level $j+p+1$ the elements of the rows of that will be summed or subtracted. Let t_0, t_1 and t_2 be the three subsets of $F_p(n-j-1)$

$$x^{(j+1)}(w,v_0) = \\ = \begin{cases} x^{(j)}(\pi_j(w_0),k) + x^{(j)}(\pi_j(w_1),k), \text{if } w \in \{t_0,t_1\} \\ \\ x^{(j)}(w_2,k) \text{if } w \in \{t_2\} \end{cases}$$

$$x^{(j+1)}(w,v_1) = \left| x^{(j)}(\pi_j(w_0),k) - x^{(j)}(\pi_j(w_1),k) \right| \quad (10)$$

$$x^{(j+1)}(w,v_2) = x^{(j)}(w,f) \quad (11)$$

Scattering order

The order m of a scattering coefficient is the number of subtractions involved in its computation. It can be demonstrated that the amplitude of a scattering coefficient of order m has a fast decay as m increases. The number of columns of the scattering coefficients depends on the scattering order m.

Since $x^{(0)} = x$, all coefficients are of order 0 for $j=0$. If $x^{(j)}(w,k)$ is of order m, then (4) and (5) imply that $x^{(j+1)}(w,v_0)$ or $x^{(j+1)}(w,v_2)$ is of order m and $x^{(j+1)}(w,v_1)$ is of order $m+1$. Meanwhile if $x^{(j)}(w,f)$ has order m, $x^{(j+1)}(w,v_2)$ has order m too. The columns of the matrix $x^{(j+1)}$, taking into consideration only the column of order $i \leq m$, are selected. Let us denote by t_i the number of coefficients of order $i \leq m$ and $m \leq [\frac{j+p}{1+p}]$ at level j. Let

$$\tilde{k} = \sum_{i=0}^{m} t_i.$$

The size of $x^{(j+1)}(w,v_l)$, with $l \in \{0,1,2\}$ is $F_p(n-j) \times \tilde{k}$.

By exploiting Zeckendorf criteria, it results that every $n \in \mathbb{N}$ can be thought as a sum of r different p-Fibonacci numbers. The number of coefficients of order m is related with the p-Fibonacci matrix, and it is equal to the number of rows with m ones, which corresponds to r. Let $B \in \mathbb{N}$. The representation of the form

$$B = \sum_{i=p}^{n-1} a_i F_p(i)$$

where $a_i \in \{0,1\}$ and $F_p(i)$ is the generalized p-Fibonacci number, is called a p-Fibonacci representation of B. The vector by coefficients, $a_p = \left[a_{n-1}, a_{n-2}, \cdots, a_p\right]_p$ is called a p-Fibonacci code of B. As previously explained, the rows of the p-Fibonacci matrix represent all $n \in \mathbb{N}$ $0 \leq n \leq N-1$.

Wavelet on graph

Let $V_{0,w} = \{w\}$ for $w \in V$. For any $j \geq 0$ and $\forall$ elements of $\{w_0\}$,$\{w_1\}$ and $\{w_2\}$ it is defined

$$V_{j+1,w} = V_{j,\pi_j(w_0)} \cup V_{j,\pi_j(w_1)} \cup V_{j,w_2}.$$

$V = \bigcup_w V_{j,w}$ defines a partition. Let V_1 and V_2 be two non-overlapping subsets of V. V_1 and V_2 are connected if at least one element of V_1 is connected to one element of V_2. In our case w_0 and w_1 are connected. A graph p-Fibonacci scattering is constructed over partition $V_{j,w}$ of V, which are obtained by aggregating vertices $V_{j+1,w} = V_{j,\pi_j(w_0)} \cup V_{j,\pi_j(w_1)} \cup V_{j,w_2}$. In the following, $\chi_{V_{j,w}}(v)$ denotes the indicator function of $V_{j,w}$ in V. Let it be

$$U_j = V_{j,w_2} \cup V_{j,\pi_j(w_0)}.$$

It follows

$$V_{j+1,w} = U_j \cup V_{j,\pi_j(w_1)}. \quad (12)$$

A p-Fibonacci wavelet computes the difference between

$$\Psi_{j+1,w} = \chi_{U_j} - \chi_{j,\pi_j(w_1)}.$$

Inner products between signals defined on V are written

$$< x, x^{'} >= \sum_{v \in V} x(v) x^{'}(v).$$

For any

$$\left\{\chi_{V_{J,w}}\right\}_{0 \leq w < F_p(n-J)} \cup \left\{\Psi_{j,w}\right\}_{0 \leq w \leq F_p(n-j), 0 \leq j < J}$$

is a family of N orthogonal p-Fibonacci wavelets which define an orthogonal basis of $\mathbb{R}^d$. It can be proved that the order $m+1$ coefficients are obtained by computing the orthogonal p-Fibonacci wavelet transform of coefficients of order m. Let $w \in \{w_0, w_1, w_2\}$, the sets of $F_p(n-j-1)$, k_m an element of order m

$$x^{(j)}(w, k_{m+1}) = \sum_{\substack{q \\ V_{j_{m+1},q} \subset V_{j,w}}} \left| < \bar{x}^{(j_m)}(\cdot, k_m), \Psi_{m+1,q} > \right| \tag{13}$$

with

$$\bar{x}^{(j_m)}(\cdot, k') = \sum_{w \in \{w_{0,m}, w_{1,m}\}} x^{(j_m)}(w, k') \chi_{V_{j_m,w}}$$

where $(w_{0,m}, w_{1,m}$ are the sets of $F_p(n-j_m))$.

If $k = \sum_{i=0}^{m} t_i$ and $j_{m+1} \geq j_m$ then $x^{(j_m)}(w, k_m)$ are coefficients of order m, whereas $x^{(j)}(w, k_{m+1})$ is a coefficient of order $m+1$. Eq. (13) proves that a coefficient of order $m+1$ is obtained by computing the wavelet transform of scattering coefficient of order m, and summing their absolute values. A coefficient of order $m+1$ thus measures the average variations of the mth scattering coefficients on neighborhoods of size k_{m+1} in the graph. To further compare the graph p-Fibonacci scattering with the classical wavelet transforms, it should be noticed that if the absolute value in (9) and (10) is removed, these equations define an orthogonal Walsh transform.

A graph p-Fibonacci scattering can thus be interpreted as an adaptive Hadamard transform over groups of vertices, which outputs positive coefficients. Walsh matrices are particular cases of Hadamard matrices.

Experimental results

The performances of the proposed method have been evaluated on two databases largely used in literature for assessing the classification performances: CIFAR-10 and MNIST. CIFAR-10 contains 60000 color images of size 32*32 pixels, while the MNIST database is composed by 70000 gray-scale images of size 28*28 pixels. The obtained results have been compared to the state-of-the-art and they are reported in Table 1 and Table 2. In those tables the percentage error of the classification is reported. It can be noticed that, if we compare the performances of algorithms based on a supervised approach, their error rate is considerably smaller. Despite of this fact, those systems need to have an additional information and processing that is not needed in the unsupervised approach. From the achieved results it can be concluded that the proposed method slightly increases the performances with respect to the reference one while requiring a lower computational complexity.

Method	Error (%)
Conv. nets (supervised)	9.8
RFL (unsupervised)	16.9
Roto-translation scattering	17.8
Graph Haar scattering	21.3
2-Fibonacci scattering	21.88
1-Fibonacci Haar scattering	21.81

Table 1: Error percentage on CIFAR-10

Method	Error (%)
Conv. nets (supervised)	0.53
RFL(unsupervised)	0.53
Roto-translation scattering	0.43
Graph Haar scattering	0.59
1-Fibonacci scattering	0.50
1-Fibonacci Haar scattering	0.49

Table 2: Error percentage on MNIST

Conclusions

In this contribution, a novel image representation framework is presented. It is based on the combination of p-Fibonacci sequences and the deep scattering networks. This approach results in a translation and rotation invariant representation of the data. This framework has been used for image classification. The performance is comparable with the those achievable by the state-of-the-art unsupervised methods.

References

[1] https://www.brandwatch.com/blog/47-facebook-statistics-2016/

[2] Xiuyuan Cheng, Xu Chen, and Stéphane Mallat. Deep haar scattering networks. *CoRR*, abs/1509.09187, 2015.

[3] Leonardo Pisano. Liber Abaci

[4] E.Zeckendorf Representation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas. *Bull. Soc. Royale des sciences de liege*, 3-4, 1972, pp. 179- 182.

[5] J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8) : 18721886, Aug 2013.